



THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

SCHOOL OF INFORMATION TECHNOLOGY AND
ELECTRICAL ENGINEERING

Thesis Project Proposal

Raymond MOGG - 44332938

Contents

1	Introduction	1
1.1	Aims	1
1.2	Scope	2
1.2.1	In scope	2
1.2.2	Dataset	2
1.2.3	Attack Types	3
1.3	Relevance	3
2	Background	3
2.1	Intrusion Detection Systems	3
2.1.1	Detection Methods	4
2.1.2	Audit Source	4
2.1.3	Behavior On Detection	5
2.2	Attacks Against Intrusion Detection Systems	6
2.2.1	Evasion Attacks	6
2.2.2	Poisoning	6
2.2.3	Denial of Service	6
2.2.4	Overstimulation	7
2.3	Genetic Algorithm	7
2.4	Related Work	8
2.4.1	Generating attacks and labelling attack datasets for industrial control intrusion detection systems	8
2.4.2	Mimicry Attacks on Host-Based Intrusion Detection Systems	9
3	Project Plan	11
3.1	Plan	11
3.2	Current Progress	13
3.3	Project Limitations	13
3.4	Project Risks	13
4	OHS Risk Assessment	15

List of Tables

1	Attack class breakdown	3
2	Pros and Cons of IDS Detection Methods	5
3	Mimicry Attack Type Summary	10
4	Project Plan Outline	11
5	Project Milestones Outline	12
6	Risk Key Table	14
7	Project Risks	14

1 Introduction

Intrusion Detection Systems (IDS) play a critical role in the current cybersecurity ecosystem by monitoring and detecting malicious attacks against computer networks and systems. [1]. While intrusion detection systems help detect and prevent malicious actors exploiting a system, it is clear that there is still plenty of work needed in the area. The average cost of a breach to a global company is \$3.86M USD, and it takes an average of 196 days to identify these breaches [2]. By investigating potential attacks against IDSes, improvements in their architecture can be made to strengthen their intrusion detection capabilities, leading to a decrease in the impact of cybersecurity attacks.

This proposal outlines an approach to completing research into the area of generating attack samples against an IDS using genetic algorithms. By leveraging the NSL-KDD dataset [3] and focusing on two attack types, the project aims to be able produce attacks that can successfully evade an IDS. It is hoped that this knowledge can then be leveraged to help improve current IDSes.

1.1 Aims

The aim of this project is to generate interpretable evasion attacks against intrusion detection systems using genetic algorithms, that will be able to evade an IDS. The IDS will be network based, and use a behavior based detection method.

In order to do this, there are a few key components that must be produced as follows:

- An IDS using decision trees in order to create a model of normal behavior (used to classify attacks / normal behavior)
- Attack samples generated using the previously produced models and genetic algorithm
- An evaluation of attack performance on the aforementioned IDS

1.2 Scope

1.2.1 In scope

The following items will be produced as part of this thesis:

- A decision tree based IDS, using behavior based detection as its detection method (**Section 2.1.1**) and utilizing a network based audit source.
- A genetic algorithm pipeline to generate attacks
- Interpretable attack samples for the two attack types, mentioned in **Section 1.2.3**
- Analysis of attack performance
- Recommendations on how intrusion detection systems can be improved to avoid these types of attacks

1.2.2 Dataset

The NSL-KDD dataset [4] will be used for producing both the attack classifier and generating attack samples. This dataset was chosen due to its wide selection of data, recent production, and extensive research surrounding it.

The NSL-KDD dataset was produced in order to solve some issues with the KDD99 dataset, such as redundant records and a low difficult level of attacks [4].

There are 22 attacks included as part of the NSL-KDD dataset [4], including - neptune, warezclient, ipsweep, portsweep, teardrop, nmap, satan, smurf, pod, back, guess_passwd, ftp_write, multihop, rootkit, buffer_overflow, imap, warezmaster, phf, land, loadmodule, spy, perl

The attacks from the NSL-KDD dataset can be broken down into the following attack classes, summarized in **Table 1** [4].

Table 1: Attack class breakdown

Attack Class	Attacks
DoS	Smurf, Land, Pod, Teardrop, Neptune, Back
R2L	Ftp_write, Guess_pass, Imap, Multihope, phf, spy, warezmaster, warezclient
U2R	Perl, buffer_overflow, Rootkit, Loadmodule
Probe	Ipsweep, nmap, portsweep, satan

1.2.3 Attack Types

In order for attacks to be effectively produced, only two attack types will be investigated. The two attack types that are being focused on are the teardrop attack and nmap probe attack. These two attacks represent two different attack scenarios - with one being a denial of service and the other being a probe, which are two commonly occurring attacks seen today [5].

1.3 Relevance

This research is highly relevant in the current cyber security ecosystem, with the number of unique cyber security attacks up 27% in 2018 [6]. By exploring how attacks can be generated that could potentially evade an IDS, measures can be put into place to improve the design of current IDS, essentially patching for these type of generated attacks. While this may be difficult, a understanding of what such generated attacks may look like will still aid in the further research of how these types of attacks may be prevented.

2 Background

2.1 Intrusion Detection Systems

IDSes exist in order to monitor various computer systems, and detect potential attacks against these systems [1]. Attacks can come both in the form of misuse by legitimate users, and in the form of malicious attacks by third parties [1]. An IDS can be broken down into 3 main components that define its area of operation. These are its detection method, behavior upon detection

and audit source location [1].

2.1.1 Detection Methods

There are two main detection methods predominantly used within current IDSes - behavior based and knowledge based detection [1]. Behavior based detection systems use pattern matching and contain a database of known attack vectors. By looking at the fingerprint of a potential attack, it can be checked against the known attack database to clarify whether this is an attack or not. This is referred to as misuse detection [1].

Behavior based detection systems work by creating a model of normal user behavior. Any behavior that falls outside this model of normal behavior is then classified as an attack. This is referred to as anomaly detection [1].

Due to the nature of the two systems, knowledge based IDS's are extremely effective at detecting known attacks, as they often contain vast amount of information about potential attacks [1]. The downfall to this is that any action by a user that does not map to a known attack is considered acceptable behavior. This means that any newly crafted attack, or a well known attack that deviates slightly from the attack pattern may be classified as normal behavior and no alarms will therefor be raised [1]. As such it is imperative that knowledge based IDS's have an up to date database of attacks.

Behaviour based IDS's on the other hand are able to detect new and unseen vulnerabilities as they simply base their detection method of the deviation from normal behavior [1]. This however has the drawback that they may produce a high false-alarm rate, due to unseen behavior that is not necessarily an attack but falls outside the normal behavior range being classified as an attack.

Table 2 highlights the pros and cons of both detection methods.

2.1.2 Audit Source

The audit source of an IDS defines where the data it processes comes from. There are two main audit source's used in current IDS's - Host based and Network based. In recent times there has also been a increase in hybrid sys-

Table 2: Pros and Cons of IDS Detection Methods

Detection Method	Positives	Negatives
Knowledge Based	<ul style="list-style-type: none"> • Low false-alarm rate • Good detection of known attacks 	<ul style="list-style-type: none"> • Poor detection of new / unknown attacks • Difficult to gather a complete database of known attacks • Maintenance of attack database can also be difficult and time consuming
Behavior Based	<ul style="list-style-type: none"> • Good detection of new / unknown attacks • Can detect “abuse of privilege” type attacks 	<ul style="list-style-type: none"> • Higher false-alarm rate • Behavior changes over time - leading to the need to retrain the behaviour model of the IDS

tems which use a combination of host based and network based audit sources in order to improve detection capabilities and performance [7].

Host based audit sources use information such as logs from host systems in order to monitor and detect potential attacks, while network based audit sources use network packets and traffic.

While host based audit sources were the first to be used [1], with an increase in network based attacks such as DoS attacks, network based audit sources are now needed in order to be able to detect a multitude of attacks [1]

2.1.3 Behavior On Detection

Behavior on detection within an IDS system is defined as the action taken once a potential attack is detected. There are two main types of possible behavior - active and inactive. Active IDS’s respond to potential attacks in a proactive manor by either logging potential attackers out of the system, or by taking corrective steps to fix the issue [1].

Inactive IDS's simply trigger alarms which are then acted upon by humans [1]. These passive systems have the trade-off that a large amount of damage may already be done before any action can be taken

2.2 Attacks Against Intrusion Detection Systems

Since IDSes are themselves a computer system, they are vulnerable to attacks and exploitation [8]. There are a few classes of attacks commonly used against IDSes.

2.2.1 Evasion Attacks

Evasion attacks are a type of attack in which an adversary carefully crafts their attack to ensure that the pattern picked up by the IDS is not classified as an attack, even though it still is malicious [8]. There are two classes of evasion attacks depending on what detection method the IDS is using. If the IDS is using a knowledge based detection system, the evasion attack aims to modify the attack enough so that the system does not register its footprint with one stored in the knowledge base. If the IDS is using a behaviour based detection system, then the evasion attack aims to mimic normal behavior despite being malicious (mimicry attack) [8].

2.2.2 Poisoning

Poisoning attacks are a quite new attack and stem from the fact that many modern IDSes use some sort of machine learning to train their models [8].

In this attack, well crafted patterns are added into the dataset used for training models such that the algorithm produced will be biased in some way [8].

2.2.3 Denial of Service

A Denial-of-Service (DoS) attack is when network traffic is used to overwhelm a system. In this an attack an attacker may employ various server weakness's or simply use pure throughput to take down services, rendering the IDS useless as the system is completely down [8].

2.2.4 Overstimulation

Overstimulation attacks can be considered similar to DoS attacks in a sense that they attempt to overwhelm the resources of a system. In an overstimulation attack, an attack generate a large number of attack patterns in an attempt to generate many attack alerts within the IDS, in turn overwhelming security operators and/or other analysers [8].

2.3 Genetic Algorithm

Genetic algorithms utilize the biological idea of combining genes to produce offspring in order to produce optimal solutions to a set problem space [9]. There are five key components within a genetic algorithm that need to be identified [9], these are:

- A genetic representation of the solution space; Each potential solution to the problem space must be able to be represented in a genetic way. This means braking the solution down into genes.
- An initial population; To begin the process an initial population of solutions is needed. The fittest will then be selected and create the next generation.
- A fitness function; A fitness function is needed in order to evaluate the quality of a given solution. This is how the best individuals are chosen to reproduce.
- A function for producing offspring; The function must take in two individuals and produce either a single, or a set of individuals. Through this process gene mutation and crossover is introduced which allows the population to evolve.
- A selection function; A function that selects the fittest offspring from the population.

By producing an algorithm that defines all of the above, attack samples can be generated and then evaluated against an IDS in order to gauge their fitness. From this the best attack samples can then be used in order to generate further attack samples.

2.4 Related Work

The following research articles are closely relate to the area of this research. An analysis has been performed on these articles to identify their scope, similarities and differences between the research, and how they can be used to guide this thesis.

2.4.1 Generating attacks and labelling attack datasets for industrial control intrusion detection systems

This PHD thesis covers three components of a framework that can be used to aid in the understanding of attacks against IDSes deployed for critical infrastructure, as well as being used to generate attack datasets to train future critical infrastructure IDSes. Of significant important to this thesis is the attack generation framework used, as a similar technique can be applied to generate IDS attacks using genetic algorithms.

In order to produce a SCADA attack generation framework, this paper utilizes a modular approach and first breaks down attack generation into ten requirements [10]. Of importance to this thesis are the following components:

- Ability to parse SCADA protocol messages
- Ability to replicate the SCADA protocol stake
- Ability to sniff local SCADA network traffic
- Ability to inject anomalous SCADA protocol messages into the network.
- Ability to modify protocol message data in real time
- Ability to flood a SCADA service with anomalous messages

While all of these items are specific to attack generation for SCADA systems, the above functionality can be generalised and then modelled towards the two attack types being investigated in this thesis and form a good basis for the development of attack generation.

2.4.2 Mimicry Attacks on Host-Based Intrusion Detection Systems

This article covers generating mimicry attacks to be used against host based IDSes, with many similarities able to be drawn to our network based approach. The article first outlines six key mimicry attack types, which are summarized in Table 3 [11]. While many of the summaries relate to host based IDS and not behavior based, they may still be applied to this thesis. For example, the insert no-ops attack may be used in a network based system by simply adding in extra network traffic, which may hide the attack as normal behavior and hence avoid detection by the IDS. The generate equivalent attacks may also be used, as there are many different ways a set of network packets can generate an attack, as seen by the many variations of DoS attacks that exist [12].

The article then goes on to detail a formal approach for generating attack samples by investigating the intersection of all sets of attacks with the intersection of the set of all sequence of system calls that do not trigger an alarm within the IDS. While this formal approach translates nicely to behavior based IDSes that may only look at the last six system calls [11], this approach does not scale to network based systems where an attack vector may consist of hundreds or thousands of network packets. As such this is where genetic algorithms will instead be used as part of this thesis, to generate samples that ultimately are able to avoid detection.

Table 3: Mimicry Attack Type Summary

Mimicry Attack Type	Summary
Slip under the radar	Avoid causing any chaos in the infiltrated application, since some IDSes only detect attack via their call signature within the program.
Be patient	Wait for a time when the attack can be executed without raising any alarms (assuming one exists).
Be patient, but make your own luck	Wait for a time when the attack can be executed without raising any alarms, but instead of simply passively waiting, nudge the application into running on the desired path for execution of the attack.
Replace system call parameters	Many IDS do not look at system call parameters. By replacing parameters, a benign system call can be made a malicious one.
Insert no-ops	Padding the attack with no-ops (operations that essentially do nothing) may class the attack as normal behavior and hence avoid detection.
Generate Equivalent Attacks	There are many ways to craft a malicious attack. By varying the attack sequence even slightly detection can be avoided.

3 Project Plan

The following outlines a general project plan as well as the related milestones for that plan. An update regarding current progress is also presented. Finally any limitations and risks associated with the project that may affect the completion of the research are outlined and a mitigation plan is presented as needed.


3.1 Plan

The following table outlines the plan for the project across the year, from the 29th July 2019 until the 30th March 2020

Table 4: Project Plan Outline

Step	Start Date	Description	Resources Required	Duration (Weeks)
1	29/7/19	Conduct background research into IDS's, IDS attacks and genetic algorithms	UQ Library	3 Weeks
2	29/7/19	Exploration and cleaning of NSL-KDD dataset for training models. Development of initial ML pipeline	NSL-KDD Dataset	2 Weeks
3	5/8/19	Project Proposal	N/A	2 Weeks
4	26/8/19	Development of initial random forests for classification of attacks. Analysis of most useful features for set attacks	NSL-KDD	3 Weeks
5	16/9/19	Seminar Preparation	Initial ML models and background research	3 Weeks
6	7/10/19	Development of initial genetic algorithm to start work on generating attacks	Initial ML models	3 Weeks
7	24/2/20	Continue work on the development of genetic algorithm and start generating attack samples	Initial ML models and genetic algorithm	8 Weeks
8	30/3/20	Compilation of all current work to begin final thesis production	Initial models, research	6 Weeks
9	30/3/20	Start work on poster and final demonstration	Initial models, research	6 Weeks

Table 5: Project Milestones Outline

Milestone	Relative Project Steps	Description	Completion Date
Project Proposal	1,2,3	Project proposal outlining the scope of the research to be conducted, as well as some required prior knowledge.	29/8/19
Initial Prototype	2,4	An initial prototype of the ML models and data pipelines to train said models.	16/9/19
Seminar Presentation	5	A seminar presenting the key content of the research and current progress.	TBD
Final Software	6,7,8	Completion of software required for generating models and attack samples.	22/6/20
Thesis Poster	9	Poster describing the thesis and work completed	3/7/20
Thesis	8	Final thesis to be presented at the end of the year	13/7/20 

3.2 Current Progress

The current progress of the project is at the third step of the project plan outline as seen in **figure 4**. Extensive background research has been conducted to gain an understanding of the required material for completion of the research, as presented in this document. Some basic data preparation and cleaning has been done on the NSL-KDD dataset. Upon approval of the plan, more cleaning and development of the initial models will begin.

3.3 Project Limitations

The following are considered potential limitations of the project, and are hence out of scope. They will not be produced as part of the thesis, but could however be investigated as further points of research.

- A full fledged IDS
- Attack samples for any other attack types besides those mentioned in **Section 1.2.4**
- A implementation of improvements for an IDS system
- A generic attack generation framework

3.4 Project Risks

The following figure outlines the risk key used to describe risks. Any risk evaluated as medium or high has been considered and a mitigation plan has been completed for these risks. Finally some limitations of the proposed project are outlined. After analysis of possible risks using the above risk matrix, the following table lists all risks requiring a mitigation plan.

Table 6: Risk Key Table

	Negligible	Minor	Moderate	Significant	Severe
Very unlikely	Low	Low	Low	Medium	Medium
Unlikely	Low	Low	Medium	Medium	Medium
Possible	Low	Medium	Medium	Medium	High
Likely	Low	Medium	Medium	High	High
Very likely	Low	Medium	High	High	High

Table 7: Project Risks

Risk	Probability	Severity	Impact	Mitigation Plan
Inability to generate attacks for either selected attack type	Unlikely	Significant	Medium	If it appears attacks may not be able to be generated for a certain attack type, have a discussion early on. Change attack types early if need be
Training times for genetic algorithms / decision trees consuming too much working time	Possible	Moderate	Medium	If training times are too high, utilize cloud computing infrastructure to run computations on more powerful machines.
Time constraints lead to incompleteness of attack generation	Unlikely	Severe	Medium	Generate attacks one attack type at a time, to ensure there is some data to analyse at all times.

4 OHS Risk Assessment

This project will be undertaken in a low risk laboratory and as such is covered by the general OHS rules, which may be found at: <https://www.itee.uq.edu.au/laboratory-rules>

References

- [1] H. Debar, M. Dacier, and A. Wespi, “Towards a taxonomy of intrusion-detection systems,” *Computer Networks*, vol. 31, no. 8, pp. 805–822, 1999.
- [2] “10 cyber security facts and statistics for 2018,” Available at <https://us.norton.com/internetsecurity-emerging-threats-10-facts-about-todays-cybersecurity-landscape-that-you-should-know.html> (2018/01/01).
- [3] I. Sharafaldin., A. H. Lashkari., and A. A. Ghorbani., “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” in *Proceedings of the 4th International Conference on Information Systems Security and Privacy - Volume 1: ICISSP*, INSTICC. SciTePress, 2018, pp. 108–116.
- [4] M. Tavallaei., E. Bagheri., W. Lu., and A. A. Ghorbani, “A detailed analysis of the kdd cup 99 data set,” *IEEE symposium on computational intelligence in security and defence applications*, 2009.
- [5] “What are the most common cyberattacks?” Available at <https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html> (2019/07/07).
- [6] P. Technologies, “Cybersecurity threatscape 2018: trends and forecasts,” Moscow, Russia, Tech. Rep., 2019.
- [7] A. Patel., M. Taghavi., K. Bakhtiyari., and J. C. Júnior, “Taxonomy and proposed architecture of intrusion detection and prevention systems for cloud computing,” in *Cyberspace Safety and Security. Lecture Notes in Computer Science, vol 7672*. Springer, 2012, pp. 441–458.
- [8] I. Corona, G. Giacinto, and F. Roli, “Adversarial attacks against intrusion detection systems: Taxonomy, solutions and open issues,” *Information Sciences*, vol. 239, p. 201, 2013.
- [9] D. L. Hudson and M. E. Cohen, “Genetic algorithms,” in *Neural Networks and Artificial Intelligence for Biomedical Engineering*. Hoboken, NJ, USA: John Wiley Sons, Inc., 2012, pp. 215–224.

- [10] N. R. Rodofile, “Generating attacks and labelling attack datasets for industrial control intrusion detection systems,” 2018. [Online]. Available: <https://eprints.qut.edu.au/121760/>
- [11] D. Wagner and P. Soto, “Mimicry attacks on host-based intrusion detection systems,” in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, ser. CCS '02. New York, NY, USA: ACM, 2002, pp. 255–264. [Online]. Available: <http://doi.acm.org/10.1145/586110.586145>
- [12] J. Pierre, “Types of dos and ddos attacks,” Available at <https://www.cybrary.it/2018/07/types-of-dos-and-ddos-attacks/> (2018/07/01).